



USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

intermediate hash

SEARCH

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used intermediate hash

Found 14,073 of 157,873

Sort results by

relevance

[Save results to a Binder](#)Try an [Advanced Search](#)Try this search in [The ACM Guide](#)

Display results

expanded form

[Search Tips](#)☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐**1 [Implementing typed intermediate languages](#)**

Zhong Shao, Christopher League, Stefan Monnier

September 1998 **ACM SIGPLAN Notices , Proceedings of the third ACM SIGPLAN international conference on Functional programming**, Volume 34 Issue 1Full text available: [pdf\(1.50 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Recent advances in compiler technology have demonstrated the benefits of using strongly typed intermediate languages to compile richly typed source languages (e.g., ML). A type-preserving compiler can use types to guide advanced optimizations and to help generate provably secure mobile code. Types, unfortunately, are very hard to represent and manipulate efficiently; a naive implementation can easily add exponential overhead to the compilation and execution of a program. This paper describes our ...

**2 [On parallel execution of multiple pipelined hash joins](#)**

Hui-I Hsiao, Ming-Syan Chen, Philip S. Yu

May 1994 **ACM SIGMOD Record , Proceedings of the 1994 ACM SIGMOD international conference on Management of data**, Volume 23 Issue 2Full text available: [pdf\(1.24 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In this paper we study parallel execution of multiple pipelined hash joins. Specifically, we deal with two issues, processor allocation and the use of hash filters, to improve parallel execution of hash joins. We first present a scheme to transform a bushy execution tree to an allocation tree, where each node denotes a pipeline. Then, processors are allocated to the nodes in the allocation tree based on the concept of synchronous execution time such that inner relations (i.e., hash tables) ...

**3 [Query evaluation techniques for large databases](#)**

Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2Full text available: [pdf\(9.37 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...



USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

intermediate hash block

SEARCH

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used intermediate hash block

Found 13,036 of 157,873

Sort results by

relevance

☒ [Save results to a Binder](#)Try an [Advanced Search](#)Try this search in [The ACM Guide](#)

Display results

expanded form

☒ [Search Tips](#)☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐1 [Parallel collision search with application to hash functions and discrete logarithms](#)

Paul C. van Oorschot, Michael J. Wiener

November 1994 **Proceedings of the 2nd ACM Conference on Computer and communications security**Full text available: [pdf\(984.91 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Current techniques for collision search with feasible memory requirements involve pseudo-random walks through some space where one must wait for the result of the current step before the next step can begin. These techniques are serial in nature, and direct parallelization is inefficient. We present a simple new method of parallelizing collision searches that greatly extends the reach of practical attacks. The new method is illustrated with applications to hash functions and discrete logarithms ...

2 [Fast joins using join indices](#)

Zhe Li, Kenneth A. Ross

April 1999 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 8 Issue 1Full text available: [pdf\(263.06 KB\)](#)Additional Information: [full citation](#), [abstract](#), [index terms](#)

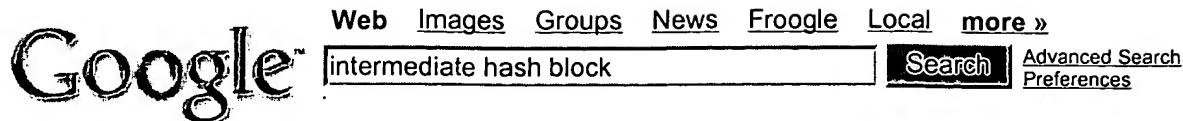
Two new algorithms, "Jive join" and "Slam join," are proposed for computing the join of two relations using a join index. The algorithms are duals: Jive join range-partitions input relation tuple ids and then processes each partition, while Slam join forms ordered runs of input relation tuple ids and then merges the results. Both algorithms make a single sequential pass through each input relation, in addition to one pass through the join index and two passes through a te ...

**Keywords:** Decision support systems, Query processing3 [Research sessions: query progress: Toward a progress indicator for database queries](#)

Gang Luo, Jeffrey F. Naughton, Curt J. Ellmann, Michael W. Watzke

June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**Full text available: [pdf\(228.58 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#)

Many modern software systems provide progress indicators for long-running tasks. These progress indicators make systems more user-friendly by helping the user quickly estimate how much of the task has been completed and when the task will finish. However, none of the existing commercial RDBMSs provides a non-trivial progress indicator for long-running queries. In this paper, we consider the problem of supporting such progress indicators.



## Web

Results 1 - 10 of about 121,000 for intermediate hash block. (0.33 seconds)

### The Whirlpool Hash Function

The W **block** cipher used by WHIRLPOOL is very similar to the AES algorithm, ...  
 The WHIRLPOOL **hashing** function is named after the Whirlpool galaxy in Canes ...  
[planeta.terra.com.br/informatica/paulobarreto/WhirlpoolPage.html](http://planeta.terra.com.br/informatica/paulobarreto/WhirlpoolPage.html) - 12k - [Cached](#) - [Similar pages](#)

### RFC1810: Report on MD5 Performance. J. Touch. June 1995. (Format:...

The first **block** is hashed with an initial seed, resulting in a **hash**. ... The 64 steps are 16 groups of 4 steps, one step per **intermediate hash** word. ...  
[rfc1810.x42.com/](http://rfc1810.x42.com/) - 18k - [Cached](#) - [Similar pages](#)

### RFC1810

A,B,C,D **intermediate hash** words X[i] input data **block** T[i] sine table lookup << i rotate i bits F logical functions of 3 args. The subscripts to X, I, ...  
[www.scit.wlv.ac.uk/rfc/rfc18xx/RFC1810.html](http://www.scit.wlv.ac.uk/rfc/rfc18xx/RFC1810.html) - 16k - [Cached](#) - [Similar pages](#)

### [doc] COBRA(tm) Consultative Products For Windows(tm)

File Format: Microsoft Word 97 - [View as HTML](#)  
 rc = **HASH**.MD5(LengthRead,2,**Block**,Digest,Work);. if (rc != 0). { . printf("\nFailed to **hash intermediate block**");. return(0); ...  
[www.secure-hash-algorithm-md5-sha-1.co.uk/manual.doc](http://www.secure-hash-algorithm-md5-sha-1.co.uk/manual.doc) - [Similar pages](#)

### Funnels

Let k1 and k2 be text **blocks**, and z the internal state of the **hash** function. ...  
 For every **intermediate** point in the mixing step, consider running the ...  
[burtleburtle.net/bob/hash/funnels.html](http://burtleburtle.net/bob/hash/funnels.html) - 13k - [Cached](#) - [Similar pages](#)

### Good Stuff: RFC 1810

When the last **block** is computed, it's "next-seed" value becomes the **hash** for ...  
 from RFC-1321 [3]): A,B,C,D **intermediate hash** words X[i] input data **block** ...  
[www.good-stuff.co.uk/useful/rfc.php?rfc=1810](http://www.good-stuff.co.uk/useful/rfc.php?rfc=1810) - 23k - [Cached](#) - [Similar pages](#)

### Yahoo! Groups : the\_gdf Messages : Message 16911 of 21912

The **hash** function is then computed iteratively using these **blocks**. Starting ...  
 the final **hash** value is not the **intermediate hash** value of a longer message. ...  
[groups.yahoo.com/group/the\\_gdf/message/16911](http://groups.yahoo.com/group/the_gdf/message/16911) - 18k - [Cached](#) - [Similar pages](#)

### RFC 1810 - Report on MD5 Performance

When the last **block** is computed, it's "next-seed" value becomes the **hash** for ...  
 from RFC 1321 [3]): A,B,C,D **intermediate hash** words X[i] input data **block** ...  
[www.rfc-archive.org/getrfc.php?rfc=1810](http://www.rfc-archive.org/getrfc.php?rfc=1810) - 33k - [Cached](#) - [Similar pages](#)

### rfc1810

This RFC uses the following notation (as from RFC-1321 [3]): A,B,C,D **intermediate hash** words X[i] input data **block** T[i] sine table lookup << i rotate i bits ...  
[ietfreport.isoc.org/idref/rfc1810/](http://ietfreport.isoc.org/idref/rfc1810/) - 19k - [Cached](#) - [Similar pages](#)

### [doc] Exam0

File Format: Microsoft Word 2000 - [View as HTML](#)  
 A **hash** index on B would return 10000 **block** references, whereas there are only ...  
 ... Underflow occurs at **intermediate** level, merging 2 **intermediate** nodes (one ...